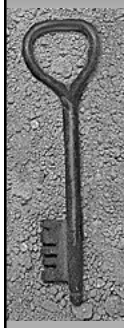
General Key Management Guidance



Key Management Policy

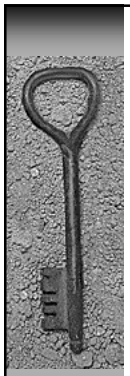
- Governs the lifecycle for the keying material
- Hope to minimize additional required documentation

Key Management Practices Statement

- Based on Key Management Policy (KMP)
- Specifies how key management procedures and techniques are used to enforce the KMP

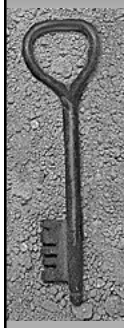
Key Usage

- A key should be used for only one purpose



Cryptoperiods

- A suitable cryptoperiod:
  - limits the amount of information protected by a given key that is available for cryptanalysis,
  - limits the amount of exposure if a single key is compromised,
  - limits the use of a particular algorithm to its estimated effective lifetime, and
  - may limit the amount of time available for cryptanalytic attacks to be useful.



Cryptoperiods (Contd.)

- Trade-offs associated with the determination of cryptoperiods involve the risk and consequences of exposure
  - A list of considerations is provided
- Discussions provided per keying material type

Domain Parameter Validation and Public Key Validation

- Domain parameters should be:
  - Generated by a trusted party, or
  - If generated by an untrusted party, should be validated by a trusted party or by the participating entities

Domain Parameter Validation and Public Key Validation (Contd.)

- Signature verification keys should be:
  - Validated for association with the private key and the owner (POP)
  - Validated by a trusted party (e.g., a CA)
- Validation of other public keys
  - Discussed in Schemes Document for key agreement
  - Guidance needed for other public keys

Compromise of Keys and Other Keying Material

- Compromise: keying material cannot be trusted to provide the required security
  - Confidentiality
  - Integrity
  - Usage or application association
  - Association with the owner or other entity
  - Association with other information

Compromise of Keys and Other Keying Material (Contd.)

- Guidance needed on limiting the consequences and recovering (when possible)
  - May need to address by key type

Accountability

- To to help prevent and to assist in mitigating the effects of a compromise
  - Used to determine when a compromise occurred and by who was involved
  - Discourages compromises by an individuals
  - Useful in recovering from a compromise

Accountability (Contd.)

- Identify
  - Keys
  - Users
  - Dates and times of use
  - Data that is protected

Audit

- To determine that procedures and practices continue to be followed
- To review and update procedures based on new technology and threats

Key Recovery

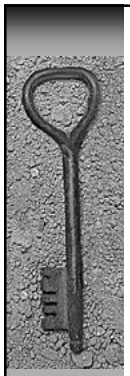
- The process of retrieving keying material from backup or archive storage when it is not otherwise available
- Purpose: to recover (e.g., decrypt) or verify (e.g., authenticate) protected information on behalf of an organization or individual
- Use or non-use of key recovery should be a conscious decision

Key Recovery (Contd.)

- Considerations for key recovery
  - Information that is stored for an extended period of time must be readily available during the lifetime of that data.
  - Transmitted information that is encrypted or authenticated may or may not require key recovery
  - Access control or authorization keys may need to be recoverable
  - Other examples?

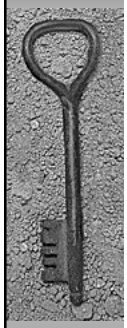
Key Recovery (Contd.)

- Key Recovery Policy (when a need for key recovery is determined)
- Define a Key Recovery System (KRS) to support the Key Recovery Policy
- Contents of the Policy (minimum):
  - What keying material needs to be saved?
  - How and where keying material is saved?
  - Who will protect the saved keying material?
  - Who can request key recovery and under what conditions?



Key Recovery (Contd.)

- Contents of the Policy (contd.)
  - How is a request authenticated and authorized?
  - Who is notified of a key recovery action?
  - How is the policy modified and by whom?
  - What audit capabilities and procedures are needed?
  - How does the KRS deal with the destruction of keying material?
  - How does the KRS deal with a compromise?



Discussion of Key Management Policy?

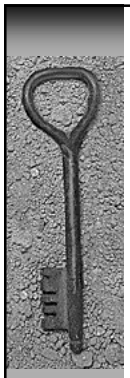
- Key Management Practices Statement
- Key Usage
- Cryptoperiods
- Domain Parameter Validation and Public Key Validation
- Compromise of Keying Material
- Accountability
- Audit
- Key Recovery

Guidance for Cryptographic Algorithm and Key Size Selection

- Approved algorithms are specified in FIPS
- Approved algorithms provide different security strengths
- In some cases, multiple key sizes are specified

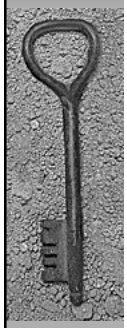
Equivalent Algorithm Strengths

- Two algorithms are considered to be of equivalent strength for the given key sizes if the amount of time needed to “break the algorithms” or determine the keys (with the given key sizes) is the same. The strength of an algorithm for a given key size is traditionally described in terms of the amount of time it takes to try all keys for a symmetric algorithm that has no short cut attacks (i.e., exhaust the key space)



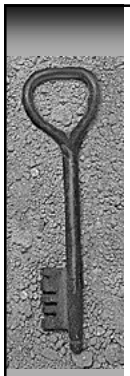
Equivalent Strengths

Bits of security	Symmetric key algs.	Hash algs.	DSA, D-H, MQV	RSA	Elliptic Curves
80		SHA-1	$L = 1024$ $N = 160$	$k = 1024$	$f = 160$
112	TDES		$L = 2048$ $N = 224$	$k = 2048$	$f = 224$
128	AES-128	SHA-256	$L = 3072$ $N = 256$	$k = 3072$	$f = 256$
192	AES-192	SHA-384	$L = 7680$ $N = 384$	$k = 7680$	$f = 384$
256	AES-256	SHA-512	$L = 15360$ $N = 512$	$k = 15360$	$f = 512$



Defining Appropriate Key Sizes

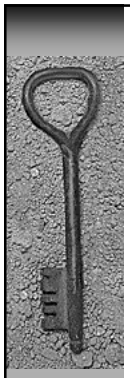
- 80 bits of security OK for now; 112 bits after 2015
  - DES key size “officially” broken in ~1997?
    - 80 bits = 24 bits more than the 56 bits of DES
    - Moore’s law: ~36 years to break an additional 24 bits
    - 1997 + 36 = 2033
  - Lenstra: 80 bits broken in 2012, assuming DES broken in 1982
  - Therefore, a conservative compromise



Defining Appropriate Key Sizes (Contd.)

Recommended algorithms and minimum key sizes

Years	Symmetric key algs. (Encryption & MAC)	Hash Alg.	HMAC	DSA, D-H, MQV	RSA	Elliptic Curves
Present - 2015	TDES AES-128 AES-192 AES-256	SHA-1 SHA-256 SHA-384 SHA-512	SHA-1 (≥80 bit key) SHA-256 (≥128 bit key) SHA-384 (≥192 bit key) SHA-512 (≥256 bit key)	Min.: L = 1024; N =160	Min.: k=1024	Min.: f=160
2016 and beyond	TDES AES-128 AES-192 AES-256	SHA-256 SHA-384 SHA-512	SHA-256 (≥128 bit key) SHA-384 (≥192 bit key) SHA-512 (≥256 bit key)	Min.: L = 2048 N = 224	Min.: k=2048	Min.: f=224



Defining Appropriate Key Sizes (Contd.)

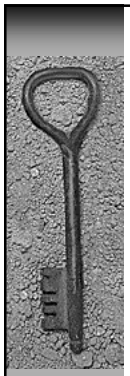
- Algorithms of different strengths and key sizes may be used together for performance, availability or interoperability reasons, provided that sufficient protection is provided
- Security provided is often equal to the weakest algorithm/key size

Defining Appropriate Key Sizes (Contd.)

- Steps in selecting the algorithm suite
  - Determine the security life of the data
  - Select algorithms and key sizes that will protect the data during its entire lifetime (using the tables and examples)
- Examples to be provided

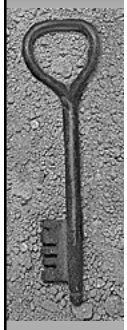
Transitioning to New Algorithms and Key Sizes

- Must address legacy systems that don't conform to the recommended algorithms and key sizes
- May not be able to "extend" the protection to the lifetime of the data (e.g., data encrypted using DES is already vulnerable)



Key Establishment Schemes

- Include additional guidance not included in the schemes document



Discussion of Algorithm Selection, Key Size Selection and Key Establishment Schemes?

- Equivalent Algorithm Strengths
- Defining Appropriate Algorithm Suites
- Transitioning to Algorithms and Key Sizes
- Key Establishment Schemes